A Fault tolerant Control Supervisory System development Procedurefor Small Satellites

The AAUSAT-II case

Izadi-Zamanabadi, Roozbeh; Larsen, Jesper Abildgaard

Publication date:
2007

Document Version
Publisher's PDF, also known as Version of record

A FAULT TOLERANT CONTROL SUPERVISORY SYSTEM DEVELOPMENT PROCEDURE FOR SMALL SATELLITES - THE AAUSAT-II CASE

Roozbeh Izadi-Zamanabadi, Jesper A. Larsen

Aalborg University, Department of Electronic Systems
Fredrik Bajers Vej 7C, DK-9220 Aalborg Ø, Denmark
E-mail:{riz,jal}@control.aau.dk, Phone +45 96 35 87 69,
Fax +45 98 15 17 39

Abstract: The paper presents a stepwise procedure to develop a fault tolerant control system for small satellites. The procedure is illustrated through implementation on the AAUSAT-II spacecraft. As it is shown the presented procedure requires expertise from several disciplines that are nevertheless necessary for obtaining a complete and consistent solution.

Keywords: Pico Satellite, Structural Analysis, Fault Tolerance, Decision Logic, AAUSAT-II

1. INTRODUCTION

Design of a complete fault tolerant supervisory control involves a number of activities. It requires methods that can help the designers to rigorously analyze the system, identify all possible/potential faults, identify the monitoring/diagnosis possibilities, design control and sensor fusion algorithms for different scenarios, design the dedicated decision logic to ensure correct decision when an event has occurred and then take the appropriate action to accommodate for the situation.

In this paper we proposed a stepwise procedure that has been employed in order to design the Fault-tolerant supervisory control for the student satellite AAUSAT-II. Due to the space constraint we limit ourselves to provide the main ideas and introduce the methods in brief in order to illustrate their applicability with help of some examples related to the AAUSAT-II control system.

[1] The authors would like to acknowledge work done by students working on the AAUSAT-II ADCS.

The paper include the following sections. In section 2 the AAUSAT-II is shortly introduced. Section 3 presents the architecture for fault tolerant control implementation. The structural analysis method in section 5 provides information on existing redundant information in the system that is useful for fault diagnosis as well as fault accommodation purposes. Fault accommodation strategies are discussed in section 6. The fault diagnosis topic, where the results of structural analysis are used, is discussed in section 7. In section 8 we discuss the fault handling methods. Then we introduce a method for designing the supervisor's decision logic in section 9.

2. AAUSAT-II DESCRIPTION

The AAUSAT-II is the second CubeSat from AAU made by students only. The CubeSat is characterized by being $100\,mm$ on each side of the cube and weighting $1000\,g$. AAUSAT II carries two science experiments, an Advanced Attitude Determination and Control System (ADCS) and a Gamma Ray Burst Detector.

2.1 Actuators and Sensors

The actuators used on the AAUSAT-II are three magnetorquers and three momentum wheels mounted perpendicular to each other and aligned to the three axes of the spacecraft frame. The momentum wheel is a small circular flywheel mounted on the drive shaft of a small DC-motor and can be used to change the attitude of the spacecraft due to the preservation of angular momentum. The magnetorquers are square coils attached to three of the sides of the spacecraft, perpendicular to each other, and exert a torque on the spacecraft by interacting with the magnetic field of the Earth, which makes them beneficial to use in LEO. In order to control the attitude with the actuators it is necessary to be able to determine the actual attitude and angular velocity. These tasks will be accomplished by combining four kinds of sensors; one tri-axis magnetometer, six gyroscopes, six sun sensors and eight temperature sensors

3. FAULT-TOLERANT CONTROL SYSTEM ARCHITECTURE

As it is illustrated in figure 1 a modular architecture for implementation of the fault tolerant control system is employed. The detector modules

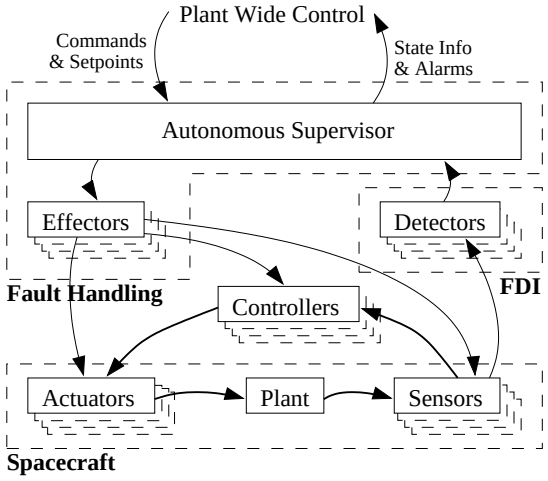

Fig. 1. An overview of fault tolerant control system architecture.

monitor the system and when a fault occurs the supervisor is informed. Based on the received information from detector modules and/or operator, the supervisor switches to appropriate state. The effector modules translate the new state and carry out the necessary changes (including changing the control strategy or activating proper sensor fusion). The modularity provides both flexibility when changes are needed and also less complicated testing procedures.

As shown on Figure 1, the spacecraft could be considered as a complex system with different sensors, actuators and controllers. Due to the system complexity as described previously it is necessary to apply a structured way of analysing the system in steps, as shown in figure 2 from (Bøgh, 1997).

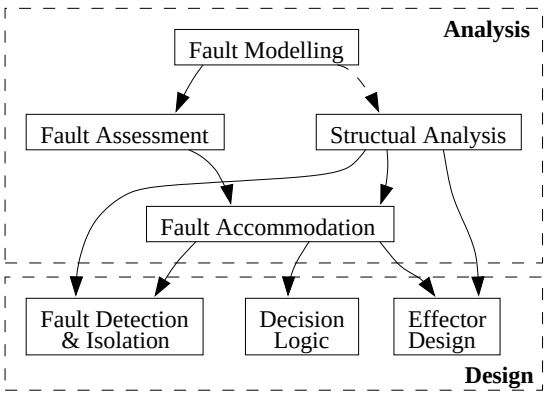

Fig. 2. Systematic fault tolerant control system development approach.

In the following we describe a stepwise procedure that is used to analyze the ADCS system with the aim of designing a fault tolerant control system for the AAUSAT-II satellite that operates in a correct and consistent manner. Different steps of this procedure are described in the sequel.

4. FAULT MODELING AND HAZARD ANALYSIS

A natural and important step in the analysis phase concerns identifying and assessing possible faults with the aim of achieving tolerance towards the most crucial ones at control system level. This step constitutes following sub-steps:

4.1 Fault modeling

This step involves dividing the system into components and analyzing each component for possible faults using a proper Hazard analysis technique. For electro-mechanical systems the Failure Mode and Effect Analysis (FMEA) technique is been used (International, 1998). The result of FMEA on the magnetorquer is shown in table 1.

4.2 Fault assessment

This step includes fault propagation analysis through the system levels, severity assessment for each fault, and causal relation analysis. The result would be a set of identified severe faults that require detection and handling.

Fault assessment is performed by using a *Severity Occurrence* (SO) index. Each fault is assessed

Table 1. FMEA for the magnetortquers.

Magnetorquers		
Produces a magnetic field to align with Earth's field		
Ref.	**Failure Effect**	**Failure Cause**
MT1	Low magnetic field	a) Broken wire / b) Bad soldering
		Component burned
MT2	Maximum magnetic field power	Shortcut to the power voltage
MT3	Wrong direction of the magnetic field	a) Misalignment of the magnetorquer caused by launch shock or wrong mounting
		b) Shortcut of some parts of the torquer to the power voltage
MT4	Wrong power of the magnetic field	Floating supply voltage caused by broken wire or bad soldering

with respect to severity and occurrence. The *severity* describes the level of impact that an occurrence of a fault has on the mission objective (here the control objective) and quantized with numbers from 1 to 10 where 10 is the highest rated i.e. severity 10 is very severe. The *occurrence* represents the probability that a fault will occur during the mission's lifetime and is quantized by numbers ranging from 10 (very likely) to 1 (highly improbable) (International, 1998).

The Severity Occurrence Index (SO) is then obtained through multiplication of the severity and occurrence values. The faults in the sensors or actuators with the highest SO Index should be considered in the design process. The SO index table for the magnetorquer is presented below.

Table 2. SO index table for the magnetortquer.

Magnetorquer				
Reference	**Severity**	**Occurrence**		**SO Index**
MT1	7	*a*	5	35
		b	4	28
MT2	10	–	3	30
MT3	3	*a*	2	6
		b	1	3
MT4	4	–	6	24

4.3 Fault simulation and injection

The faults in the SO index tables are evaluated by means of either simulation or fault injection on the actual system. Since our system is a satellite, fault injection on the real system is for large number of faults is not a realistic option. Hence, faults are modeled and simulated on a computer model and evaluated through studying their effect on the control objectives. Figure 3 shows the effect of a winding break in the x-axis magnetorquer at t=100s, which unhandled would result in a diverging attitude relative to the desired.

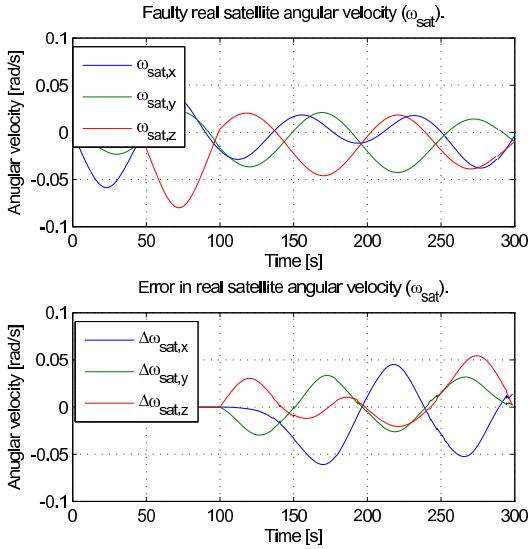

Fig. 3. Fault injection example. The top figure shows how the satellite attitude evolves and the bottom figure shows how the attitude diverges from the non-faulty attitude.

5. STRUCTURAL ANALYSIS

A general framework for an analysis of diagnostic feasibility and possibility is the structural approach (Declerck and Staroswiecki, 1991; Izadi-Zamanabadi and Staroswiecki, 2000; Blanke *et al.*, 2003). The main objective of the structural approach is to identify the parts/ subsystems in the plant that contain redundant information. The redundant information can then be analyzed and used for diagnosing faults by using appropriate methods. The structure model of a system does not depend on detailed knowledge of parameters or dynamic relations within the plant, only relation between the constraints (i.e. diff. equations, algebraic eqs., rules) and the variables are considered. It shall be noticed that the analysis is performed on the complete nominal system (with no fault). In addition, the structural analysis is used to identify the sensor fusion possibilities in the system, which then is employed to accommodate for different faulty sensors. The structural analysis method will be partially illustrated in the following (For more information please consult (Izadi-Zamanabadi, 2002). The dynamic and kinematics of the satellite is described by the following equations:

$$\begin{bmatrix}\dot{\omega}\\ \dot{\mathbf{q}}\end{bmatrix} = \begin{bmatrix}\mathbf{I}^{-1}\mathbf{N_{ext}} + \mathbf{N_{ctrl}} - \mathbf{S}(\omega)(\mathbf{I}\omega + \mathbf{h_{mw}})\\ \frac{1}{2}\mathbf{\Omega}\cdot\mathbf{q}\end{bmatrix}$$

$$\begin{bmatrix}\omega_{\mathbf{m}}\\ \mathbf{q_m}\end{bmatrix} = \begin{bmatrix}\mathbf{1_{3\times 3}} & \mathbf{0_{3\times 4}}\\ \mathbf{0_{4\times 3}} & \mathbf{1_{4\times 4}}\end{bmatrix}\begin{bmatrix}\omega\\ \mathbf{q}\end{bmatrix}$$

Where ω is the angular velocity vector, $\mathbf{q}$ is the quaternion vector, $\mathbf{h_{mw}}$ is the impulse of the momentum wheels, $\mathbf{N_{ext}}$ is the external torques, i.e. disturbances and $\mathbf{N_{ctrl}}$ is the net control

torque from the magnetorquers and the momentum wheels; $\mathbf{N_{ctrl}} = \mathbf{N_{mt}} - \mathbf{N_{mw}}$.

To illustrate the method we focus on the equations given by parts of the kinematics and measurements. Looking at the first row of the differential equation describing the kinematics we get:

$$\dot{q}_1 = \frac{1}{2}(\omega_3.q_2 - \omega_2.q_3 + \omega_1.q_4)$$

In structural analysis formulation this equation is represented by a constraint of following form:

$$c_1(\dot{q}_1, \omega_3, q_2, \omega_2, q_3, \omega_1, q_4) = 0,$$

Correspondingly, other constraints are define:

$$c_2\,(q_1, \dot{q}_2, q_3, q_4, \omega_1, \omega_2, \omega_3) = 0$$
$$c_3\,(q_1, q_2, \dot{q}_3, q_4, \omega_1, \omega_2, \omega_3) = 0$$
$$c_4\,(q_1, q_2, q_3, \dot{q}_4, \omega_1, \omega_2, \omega_3) = 0$$

In addition, differential constraints are introduced to indicate the differential terms in the equations (see (Blanke *et al.*, 2003) for more details):

$$d_1(\dot{q}_1, q_1) = 0, \quad d_2(\dot{q}_2, q_2) = 0,$$
$$d_3(\dot{q}_3, q_3) = 0, \quad d_4(\dot{q}_4, q_4) = 0$$

The measurements are represented by the following constraints:

$$c_5(q_{1m}, q_1) = 0, \quad c_6(q_{2m}, q_2) = 0,$$
$$c_7(q_{3m}, q_3) = 0, \quad c_8(q_{4m}, q_4) = 0,$$
$$c_9(\omega_{1m}, \omega_1) = 0, \quad c_{10}(\omega_{2m}, \omega_2) = 0,$$
$$c_{11}(\omega_{3m}, \omega_3) = 0$$

The structural model is obtained by establishing all constraints in the system. The structural model is then represented by a directed graph, which illustrates the link between different variables and constraints. It will also provide a straight way to calculate the unknown variables from known variables through the involved constraints. For instance, the following figure depicts the mentioned constraints where the gyroscopes' measurements are involved. The redundant information in the system is then identified by means of matching. For instance, ω_1 is matched to c_9 and indicated by $\omega_1 \leftrightarrow c_9$, which means that the value of ω_1 can be calculated through c_9 when the other involved variables, here ω_{1m}, are known. Since we have 15 constraints and 11 unknown variables, we can at most obtain 11 matched pairs, as illustrated in table 3:

Table 3. Possible match for gyroscope system

$c_5 \leftrightarrow q_1$	$c_6 \leftrightarrow q_2$	$c_7 \leftrightarrow q_3$	$c_8 \leftrightarrow q_4$
$c_9 \leftrightarrow \omega_1$	$c_{10} \leftrightarrow \omega_2$	$c_{11} \leftrightarrow \omega_3$	$d_1 \leftrightarrow \dot{q}_1$
$d_2 \leftrightarrow \dot{q}_2$	$d_3 \leftrightarrow \dot{q}_3$	$d_4 \leftrightarrow \dot{q}_4$	

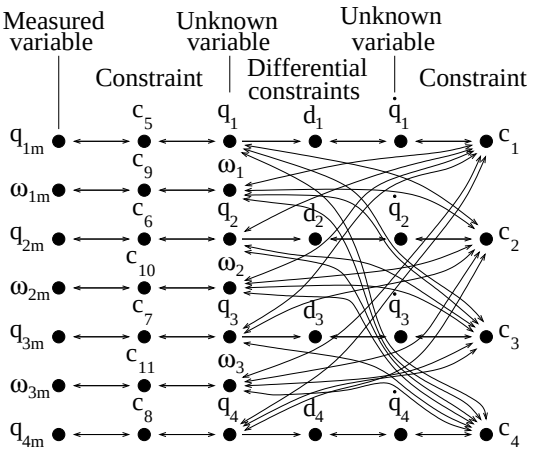

Fig. 4. A structural digraph

The resulting 4 unmatched constraints, i.e. $c_1 - -c_4$, represent redundant information in the system and can be used for fault detection as well as fault accommodation purposes.

6. FAULT ACCOMMODATION STRATEGIES

In this step the strategies to handle different selected faults will be handled where for every operational mode, the required subsystems and instrumentations are evaluated w.r.t. possible redundancies in order to maintain the operation. In addition, critical time requirements for handling the faults in order to avoid loss of control objectives are also established in this step. For the sensor faults, the fault accommodation strategy is heavily based on the sensor fusion possibilities and uses the redundant information that is provided by structural analysis. The actuator faults are mostly handled through control reconfiguration or hardware redundancy.

The mentioned steps provide a comprehensive method for analysis a system with the aim of achieving fault tolerant. The final outcome of this phase is a list of severe faults that need to be detected and handled and a clear view of the means by which they should be handled.

The next phase is the detailed phase where the aim is to materialize the detection and handling activities in a coherent and consistent manner. This phase consist of three steps: developing algorithms to detect identified faults, Developing decision logic that react to the possible fault (or commands) and determine a corresponding logical state, and finally, algorithms/procedures that interpret the new state and provide the required corresponding functionality.

7. FAULT DETECTION AND ISOLATION

Depending on the HW/SW redundancies and the complexity of the underlying system dynamics

there exist various methods and algorithms for fault diagnosis purposes.
Fault detection and isolation could benefit from the SA by utilizing redundant information, provided by SA, and employing model-based methods such as observer-based or parity space methods. To illustrate the idea, we would like to generate a residual that illustrated a fault in the gyro measuring ω_1. Using direct method we can establish a residual signal (as an indicator for faults) as follows:

$$r_1 = \omega_{1m} \cdot q_{4m} - 2 \cdot \frac{dq_{1m}}{dt} + \omega_{3m} \cdot q_{2m} - \omega_{2m} \cdot q_{3m}$$

It should be noted that in order to obtain sufficiently good estimate it can be necessary to filter the measurements. This subject is out of the scope of this paper and will not be considered further.

By manipulating all redundant information in the system through development of dedicated Fault diagnosis algorithms, the possibility of detecting and isolating all chosen severe faults is then evaluated.

If all faults can be detected and isolated, then the next step will be initiated. Otherwise, the development group has to find a negotiated solution between adding additional instrumentation or compromise to a lower level of FTC requirements. In the case where we add additional instrumentations (sensors), previous steps of analysis phase must be repeated.

8. FAULT HANDLING AND ACCOMMODATION

Two cases have been of interest: faults in sensors and faults in actuators. In the sensor case fault handling is achieved either through software or hardware redundancy. For Pico-satellites, as AAUSAT-II, the possibility of using HW redundancy is very limited, hence software redundancy is the preferred one. The use of SA results becomes again very beneficial as it is illustrated by the following example: Assume that a fault in the gyro that provides measurement ω_{1m} is faulty. We can calculate/estimate the value of ω_1 through the unmatched constraint c_1 (c_2, c_3, c_4 can also be used):

$$\hat{\omega}_1 = \frac{2 \cdot \frac{dq_{1m}}{dt} - \omega_{3m} \cdot q_{2m} + \omega_{2m} \cdot q_{3m}}{q_{4m}}$$

As mentioned in the previous section, the numerical problems can be handled by introducing appropriate (low-, band-pass) filters. The most probable solution in case of actuator failures is either HW redundancy or control reconfiguration. For AAUSAT-II case the *pointing* mode is achieved either by using both magnetorquers and momentum wheels (indicated by a submode called *Fine Pointing* (Fine) mode) or magnetorquers only (indicated by a submode called *Eco Pointing* (Eco) mode). The Eco mode provides a lesser degree of pointing accuracy; In case of failure in momentum wheels, the Eco mode, with its dedicated control algorithm, will be activated.

All fault handling and accommodation strategies are activated by *effector* modules in figure 2 depending on the current state of the supervisor's decision logic.

9. DECISION LOGIC

The process starts will defining the mission objectives and the set of mission modes in which different the mission objectives can be achieved. For instance, the mode "pointing" is defined for the case where we would like to initiate the scientific tasks of the mission. For each mission mode, different control modes can be defined. These control modes reflect the set of instrumentation (sensors/actuators and control algorithms) by which the objectives for the mission phase are achieved albeit degraded performance.

Figure 5 illustrates the conceptual arrangement of the modes (operation/control) based on which the decision logic is constructed. The priority of the modes are illustrated in the figure as well. For instance, if the control objective in the "Fine" mode cease to be reachable then the control mode will change to the less demanding mode (in this case the "Eco" mode), If no control mode is left then the mission mode will be changed to the less demanding mission mode, etc.. Thus if everything fails the system will finally reach idle mode where the ADCS will stop all control actions and merely return house keeping information from available sensors.

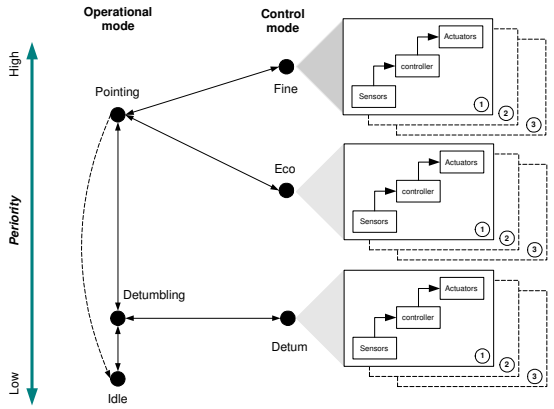

Fig. 5. A graphical representation of the different modes in the decision logic.

As mentioned before each control mode involves a combinations of sensors, actuators and controller (algorithms) to fulfill the given control objectives

in that mode. Decision on which combination of these should be used depends on the *Health Condition*, HC, of the involved components. The boolean valued health condition will be provided by dedicated fault diagnosis modules, i.e. if HC_x is true then the component x is fault free otherwise it is false. A boolean string for each feasible transducer combination can now be made for an appropriate controller (algorithm). To maintain clarity and overview, similar transducers can be grouped together since they are often used together, i.e. magnetometers.

Through design and simulation the (sub)set of actuators needed for achieving control objectives with various degree of performance need to be determined. For instance, analysis and simulations show that the *Detum* controller still can meet the control objectives with one magnetorquer failing. In addition, some sensors, when failed, can be replaced through sensor fusion measures which uses the Structural analysis results (as described in the previous section).

An example of generating boolean strings which will satisfy the control objectives in *Detum* modes, albeit with varying performance is given below.

$$\begin{array}{l}(a)\ HC_{mt_1} \wedge HC_{mt_2} \wedge HC_{mt_3}\\ (b)\ HC_{mt_1} \wedge HC_{mt_2}\\ (c)\ HC_{mt_1} \wedge HC_{mt_3}\\ (d)\ HC_{mt_2} \wedge HC_{mt_3}\end{array}$$

Since it is likely that more than one string would results in true the strings are prioritized through the "$\succ$" operator, which, for the above example, gives

$$S_{mt} = (a) \succ ((b) \vee (c) \vee (d))$$

Which is to be read as: Take (a) whenever possible. If not possible then take either (b), (c) or (d) (notice that they can not be true at the same time).

Now the building of a mode in the supervisor is done by combining sets of transducers and activating the associated controller. For the detumbling controller, *Detum*, case it would look like the following:

$$Detum = S^*_{mt} \wedge (S^*_B \vee S^*_G)$$

Where the "*" denotes the string of choice from the set of strings, thus the *Detum* is to be read as: Detumbling mode is possible if at least two magnetorquers are working and either the magnetometers, S_B, or the rate gyros, S_G are working.

In the same manner, we set up the mission modes:

$$Mission = Pointing \succ Detumbling \succ Idle$$

where

$$Pointing = Fine \succ Eco$$

The Supervisor complexity is then reduced to a test of a boolean expression to make sure that the wanted mode can be fulfilled and then to pass the logic construction string to the appointed effector.

Note: The introduced logic design method provides a completely autonomous switching between different mission as well as control modes. The operator interference (in form of commands) in the logic will be facilitate by simply introducing a logic value that facilitates a forced switch between modes. For instance, the mission can be redefined to

$$\begin{aligned}Mission = &(Pointing \ \wedge \ OP_{Point}) \succ\\ &(Detumbling \ \wedge \ OP_{Detum}) \succ Idle\end{aligned}$$

Hence, if the operator wishes to switch from the Pointing mode to Idle mode he can set the values of OP_{Point} and OP_{Detum} to false and force the ADCS system to operate in the Idle mode.

10. CONCLUSION

In this paper we have provided a stepwise procedure for analysis and design of a fault-tolerant supervisory control system for small satellites. Carrying through different steps requires tools/methods from various disciplines such as Hazard analysis, modeling and control, etc. Our attempt has focused on using methods that can be carried through preferably by using automatic software tools in order to minimize the human generated errors in the design. Based on our experience, we expect that the presented procedure facilitates a time efficient solution for developing a FTC supervisory system.

REFERENCES

Blanke, M., M. Kinnaert, J. Lunze and M. Staroswiecki (2003). *Diagnosis and Fault-tolerant Control*. Springer-Verlag.

Bøgh, S.A. (1997). Fault tolerant Control Systems. PhD thesis. Aalborg University, Denmark.

Declerck, P. and M. Staroswiecki (1991). Characterization of the canonical components of a structural graph for fault detection in large scale industrial plants. In: *Proceedings of ECC*. pp. 298–303.

International, Quality Associates, Ed. (1998). *FMEA - Quick Refrence Guide*. Quality Associates International. `http://www.quality-one.com/services/fmea.cfm`.

Izadi-Zamanabadi, R. (2002). Structural analysis approach to fault diagnosis with application to fixed-wing aircraft motion. In: *American Control Conference*.

Izadi-Zamanabadi, R. and M. Staroswiecki (2000). A structural analysis method formulation for fault-tolerant control system design. In: *39th IEEE Conference on Decision and Control*. pp. 4901–4902.